

```
C      SUBROUTINE QLRPS(FMHZ,ZSYS,EN0,IPOL,EPS,SGM)
C          PREPARES PARAMETERS
C          SETS--
C          WN,ENS,GME,ZGND
C
C          COMMON/PROP/KWX,AREF,MDP,DIST,HG(2),WN,DH,ENS,GME,ZGND,
C          HE(2),DL(2),THE(2)
C          COMPLEX ZGND
C
C          COMPLEX ZQ
C
C          DATA GMA/157E-9/
C
C          WN=FMHZ/47.7
C          ENS=EN0
C          IF(ZSYS .NE. 0.) ENS=ENS*EXP(-ZSYS/9460.)
C          GME=GMA*(1.-0.04665*EXP(ENS/179.3))
C          ZQ=CMPLX(EPS,376.62*SGM/WN)
C          ZGND=CSQRT(ZQ-1.)
C          IF(IPOL .NE. 0) ZGND=ZGND/ZQ
C          RETURN
C          END
```

```

SUBROUTINE QLRA(KST,KLIMX,MDVARX)
  DIMENSION KST(2)
  PREPARES THE LONGLEY-RICE MODEL IN THE AREA PREDICTION MODE
C
C
  COMMON/PROP/KWX,AREF,MDP,DIST,HG(2),WN,DH,ENS,GME,ZGND,
X   HE(2),DL(2),THE(2)
  COMPLEX ZGND
  COMMON/PROPV/LVAR,SGC,MDVAR,KLIM
C
  DO 10 J=1,2
    IF(KST(J)-1) 11,12,13
  11   HE(J)=HG(J)
    GO TO 15
  12   Q=4.
    GO TO 14
  13   Q=9.
  14   IF(HG(J) .LT. 5.) Q=Q*SIN(0.3141593*HG(J))
      HE(J)=HG(J)+(1.+Q)*EXP(-AMIN1(20.,2.*HG(J)/AMAX1(1E-3,DH)))
  15   Q=SQRT(2.*HE(J)/GME)
      DL(J)=Q*EXP(-0.07*SQRT(DH/AMAX1(HE(J),5.)))
      THE(J)=(0.65*DH*(Q/DL(J)-1.)-2.*HE(J))/Q
  10  CONTINUE
C
  MDP=1
  LVAR=MAX0(LVAR,3)
  IF(MDVARX .LT. 0) GO TO 21
    MDVAR=MDVARX
    LVAR=MAX0(LVAR,4)
  21  IF(KLIMX .LE. 0) GO TO 22
    KLIM=KLIMX
    LVAR=5
  22  CONTINUE
  RETURN
END

```

```

SUBROUTINE QLRPFL(PFL,KLIMX,MDVARX)
  DIMENSION PFL(5)

C      SETS UP AND RUNS THE LONGLEY-RICE MODEL IN THE POINT-TO-POINT
C      MODE USING THE TERRAIN PROFILE IN PFL.
C      PFL(1)=ENP, PFL(2)=XI, PFL(3)=Z(0),...
C

COMMON/PROP/KWX,AREF,MDP,DIST,HG(2),WN,DH,ENS,GME,ZGND,
X   HE(2),DL(2),THE(2)
      COMPLEX ZGND
COMMON/PROPV/LVAR,SGC,MDVAR,KLIM
C      DIMENSION XL(2)
C
      DIST=PFL(1)*PFL(2)
      NP=PFL(1)
      CALL HZNS(PFL)
C      FIND DELTA H
      DO 11 J=1,2
11    XL(J)=AMIN1(15.*HG(J),0.1*DL(J))
      XL(2)=DIST-XL(2)
      DH=DLTHX(PFL,XL(1),XL(2))
C      FIND EFFECTIVE HEIGHTS HE
      IF(DL(1)+DL(2) .LT. 1.5*DIST) GO TO 25
C      LINE-OF-SIGHT
      CALL ZLSQ1(PFL,XL(1),XL(2),ZA,ZB)
      HE(1)=HG(1)+DIM(PFL(3),ZA)
      HE(2)=HG(2)+DIM(PFL(NP+3),ZB)
      DO 21 J=1,2
21    DL(J)=SQRT(2.*HE(J)/GME)*EXP(-0.07*SQRT(DH/AMAX1(HE(J),5.)))
      Q=DL(1)+DL(2)
      IF(Q .GT. DIST) GO TO 23
      Q=(DIST/Q)**2
      DO 22 J=1,2
      HE(J)=HE(J)*Q
22    DL(J)=SQRT(2.*HE(J)/GME)*EXP(-0.07*SQRT(DH/AMAX1(HE(J),5.)))
23    GO TO 28
C      TRANSHORIZON
25    CALL ZLSQ1(PFL,XL(1),0.9*DL(1),ZA,Q)
      CALL ZLSQ1(PFL,DIST-0.9*DL(2),XL(2),Q,ZB)
      HE(1)=HG(1)+DIM(PFL(3),ZA)
      HE(2)=HG(2)+DIM(PFL(NP+3),ZB)
28    CONTINUE
C
      MDP=-1
      LVAR=MAX0(LVAR,3)
      IF(MDVARX .LT. 0) GO TO 31
      MDVAR=MDVARX
      LVAR=MAX0(LVAR,4)
31    IF(KLIMX .LE. 0) GO TO 32
      KLIM=KLIMX
      LVAR=5
32    CONTINUE

```

```
C          SUBROUTINE LRPROP(0.)  
C  
C          RETURN  
END
```

```

FUNCTION DLTHX(PFL,X1,X2)
DIMENSION PFL(5)

C      COMPUTES THE TERRAIN IRREGULARITY PARAMETER DH FROM THE
C      PROFILE PFL BETWEEN POINTS AT X1 .LT. X2.

C      DIMENSION S(247)

C
NP=PFL(1)
XA=X1/PFL(2)
XB=X2/PFL(2)
DLTHX=0.
IF(XB-XA .LT. 2.) GO TO 80
KA=0.1*(XB-XA+8.)
KA=MIN0(MAX0(4,KA),25)
N=10*KA-5
KB=N-KA+1
SN=N-1
S(1)=SN
S(2)=1.
XB=(XB-XA)/SN
K=XA+1.
XA=XA-FLOAT(K)
DO 10 J=1,N
11   IF(XA .LE. 0.) GO TO 12
     IF(K .GE. NP) GO TO 12
     XA=XA-1.
     K=K+1
     GO TO 11
12   S(J+2)=PFL(K+3)+(PFL(K+3)-PFL(K+2))*XA
10   XA=XA+XB
     CALL ZLSQ1(S,0.,SN,XA,XB)
     XB=(XB-XA)/SN
     DO 15 J=1,N
       S(J+2)=S(J+2)-XA
15   XA=XA+XB
C
     DLTHX=QTILE(N,S(3),KA)-QTILE(N,S(3),KB)
     DLTHX=DLTHX/(1.-0.8*EXP(-AMIN1(20.,(X2-X1)/50E3)))
80   RETURN
END

```

```

SUBROUTINE HZNS(PFL)
DIMENSION PFL(5)

C
C      TO FIND HORIZONS FROM ANTENNAS WITH HEIGHTS HG AT THE TWO
C      ENDS OF THE PROFILE PFL.
C      PFL(1)=ENP, PFL(2)=XI, PFL(3)=Z(0),...
C      OUTPUT--DISTANCES DL, TAKE-OFF ANGLES THE.
C      DL=DIST IF THE PATH IS LINE OF SIGHT
C

COMMON/PROP/KWX,AREF,MDP,DIST,HG(2),WN,DH,ENS,GME,ZGND,
X   HE(2),DL(2),THE(2)
      COMPLEX ZGND

C
LOGICAL WQ
C
NP=PFL(1)
XI=PFL(2)
ZA=PFL(3)+HG(1)
ZB=PFL(NP+3)+HG(2)
QC=0.5*GME
Q=QC*DIST
THE(2)=(ZB-ZA)/DIST
THE(1)=THE(2)-Q
THE(2)=-THE(2)-Q
DL(1)=DIST
DL(2)=DIST
IF(NP .LT. 2) GO TO 18
SA=0.
SB=DIST
WQ=.TRUE.
DO 10 I=2,NP
  SA=SA+XI
  SB=SB-XI
  Q=PFL(I+2)-(QC*SA+THE(1))*SA-ZA
  IF(Q .LE. 0.) GO TO 11
    THE(1)=THE(1)+Q/SA
    DL(1)=SA
    WQ=.FALSE.
11  IF(WQ) GO TO 10
  Q=PFL(I+2)-(QC*SB+THE(2))*SB-ZB
  IF(Q .LE. 0.) GO TO 10
  THE(2)=THE(2)+Q/SB
  DL(2)=SB
10  CONTINUE
C
18  RETURN
END

```

```

FUNCTION QTILE(NN,A,IR)
DIMENSION A(NN)

C
C      REORDERS A SO THAT A(J),J=1...IR ARE ALL .GE.
C      ALL A(I),I=IR...NN. IN PARTICULAR, A(IR) WILL HAVE THE SAME
C      VALUE IT WOULD HAVE IF A WERE COMPLETELY SORTED IN
C      DESCENDING ORDER.
C      RETURNS QTILE=A(IR)
C

M=1
N=NN
K=MIN0(MAX0(1,IR),N)
10  CONTINUE
    Q=A(K)
    I0=M
    J1=N
11  CONTINUE
    DO 12 I=I0,N
        IF(A(I) .LT. Q) GO TO 13
12  CONTINUE
    I=N
13  J=J1
    DO 14 JJ=M,J1
        IF(A(J) .GT. Q) GO TO 15
14  J=J-1
    J=M
15  IF(I .GE. J) GO TO 16
    R=A(I)
    A(I)=A(J)
    A(J)=R
    I0=I+1
    J1=J-1
    GO TO 11
16  IF(I .GE. K) GO TO 17
    A(K)=A(I)
    A(I)=Q
    M=I+1
    GO TO 10
17  IF(J .LE. K) GO TO 20
    A(K)=A(J)
    A(J)=Q
    N=J-1
    GO TO 10
20  QTILE=Q
    RETURN
END

```

```

SUBROUTINE ZLSQ1(Z,X1,X2,Z0,ZN)
DIMENSION Z(5)

C
C      LINEAR LEAST SQUARES FIT BETWEEN X1, X2 TO THE FUNCTION
C      DESCRIBED BY Z--
C      Z(1)=EN, NUMBER OF INTERVALS, Z(2)=XI, INTERVAL LENGTH,
C      Z(J+3), J=0,...,EN, FUNCTION VALUES.
C      OUTPUT-- VALUES OF THE LINE, Z0 AT 0, ZN AT XT.
C

C      XN=Z(1)
C      XA=AINT(DIM(X1/Z(2),0.))
C      XB=XN-AINT(DIM(XN,X2/Z(2)))
C      IF(XB .GT. XA) GO TO 1
C      XA=DIM(XA,1.)
C      XB=XN-DIM(XN,XB+1.)
1     JA=XA
JB=XB
N=JB-JA
XA=XB-XA
X=-0.5*XA
XB=XB+X
A=0.5*(Z(JA+3)+Z(JB+3))
B=0.5*(Z(JA+3)-Z(JB+3))*X
IF(N .LT. 2) GO TO 11
DO 10 I=2,N
JA=JA+1
X=X+1.
A=A+Z(JA+3)
B=B+Z(JA+3)*X
10    CONTINUE
11    A=A/XA
B=B*12./((XA*XA+2.)*XA)
Z0=A-B*XB
ZN=A+B*(XN-XB)
RETURN
END

```